(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 February 2006 (09.02.2006)

PCT

(10) International Publication Number
WO 2006/015168 A2

(51) International Patent Classification:
H04L 12/26 (2006.01)

(21) International Application Number:
PCT/US2005/026887

(22) International Filing Date: 28 July 2005 (28.07.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/592,232	28 July 2004 (28.07.2004)	US
11/191,493	27 July 2005 (27.07.2005)	US

(71) Applicant *(for all designated States except US)*: **AUDIBLE MAGIC CORPORATION** [US/US]; 985 University Avenue #35, Los Gatos, CA 95032 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **SCHREMPP, James, B.** [US/US]; 14587 Oak Street, Saratoga, CA 95070 (US).

(74) Agent: **WILBAR, William, P.**; Sierra Patent Group, Ltd., P.O. Box 6149, Stateline, NV 89449 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

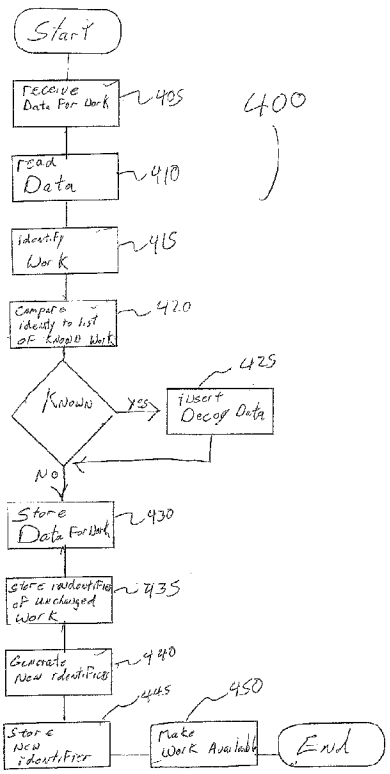
(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: SYSTEM FOR DISTRIBUTING DECOY CONTENT IN A PEER TO PEER NETWORK



(57) Abstract: A system for generating and distributing decoy data for a work in a Peer to Peer network. The system is included in software providing a Peer to Peer connection for a processing system. The processing system receives data for a work over a Peer to Peer connection. The system then determines whether, the work is a protected work. If the work is a protected work, the system insert decoy data into the data of the work and stores the data of the work including the decoy data in memory. The data of the work including the decoy data is then made available for transfer over the Peer to Peer network by the processing system.

WO 2006/015168 A2

SYSTEM FOR DISTRIBUTING DECOY CONTENT IN A PEER TO PEER NETWORK

FIELD OF THE INVENTION

This invention relates to Peer to Peer connections of processing devices over a network. More particularly, this invention relates to the creation and distribution of decoy content to discourage unauthorized transfers of protected works. Still more particularly, this invention relates to a system where Peer to Peer software in a processing device detects a transfer of a protected work to the processing device and generates decoy data in the received data of the work for storage in the processing device.

BACKGROUND

There are many so called "Peer to Peer" (P2P) products available for connecting processing devices over a network to share data between the processing devices. The P2P products are typically software that are executed by the processing device to provide a connection between the processing devices. The products allow a user to designate a collection of files that are to be "shared." That is, these files are made available for retrieval by other processing devices executing the P2P software. The P2P products also allow a user to locate files of interest on another processing system and to retrieve files from the other processing system. Typical P2P products run on top of vastly interconnected and loosely coupled networks. These networks run on top of typical networks like TCP, UDP, ATM, etc.

Intellectual property right holders of a work typically have a right to payment when the work is used. For purposes of the present discussion, a work is anything that is fixed in a tangible medium. Some examples of works include, but are not limited to, audio renderings, video renderings, images, video/audio renderings, and software. An example of an audio rendering include a song and/or other audio track. Examples of video renderings include an animation and/or a video sequence. Examples of an image include a photograph and/or paintings. Examples of audio/video renderings include movies, television shows, and cartoons. Examples of software include word processing programs and video games.

Most works have a property right, such as a copyright associated with a work. Thus, the owner of the property right is entitled to a royalty or other form of compensation for use of the work. For example, an owner of a song copyright, such as a songwriter, is entitled to a royalty for each copy of the recording produced.

Courts today have sometimes found that the transfer of a work over a network constitutes a use of that work and thus a license is required from the copyright holder and some form of payment is typically due.

The files that are transferred over Peer to Peer "P2P" networks often are the embodiment of intellectual property of a work i.e. a copyright protected work. The transfer of files containing protected works is often done without the permission of the owner of the right.

Some copyright holders have taken objection to P2P file transfers and have taken steps to disrupt these transfers. As one example, a right holder might become a peer in one of these networks. This peer will share what appears to be a copy of a desirable work. However, that copy will in fact be defective in some way. The copy might be damaged or might even contain other content that the right holder wishes to convey. Damaged or substitute content is called "decoy" content.

The right holders might take aggressive steps to introduce these damaged files into these P2P networks. The right holders might not act as just one peer on the network, but act as hundreds or thousands, or hundreds of thousands of peers, thus flooding the network with damaged files. This action is called "spoofing." Their intent is to make the network unreliable for the transmission of their works. Their hope is that an unreliable network will discourage users from transferring these works.

However, as aggressive as this sounds, the approach is not effective. P2P products have become adept at sharing information between peers. When one peer finds that decoy content is being shared by a particular IP address, it alerts other peers and the other peers in turn alert others. The source IP address is quickly eliminated from the network and the spoofer becomes ineffective. Some networks are able to eliminate entire IP address ranges from their scope. Some P2P products use a hash code or other mechanism to verify that the downloaded content is not damaged and reject any work with damaged content.

Thus, there is a need in the art for a method of introducing decoy material into a P2P network that can defeat the detect of the spoofed material in order to allow propagation of the spoofed material through a P2P network to discourage unauthorized transfers over the network of a protected work.

SUMMARY OF THE INVENTION

The above and other problems are solved and an advance in the art is made by the decoy data generation and distribution system for a P2P network in accordance with this invention. The main advantage of this invention is that a system is provided in which decoy data is inserted into a file including valid data of a protected work. Thus, the decoy is distributed over the network with identification data of a valid copy of a protected work. This makes the detection and elimination of decoy data from a P2P network harder and encourages authorized transfers of a protected work.

In accordance with this invention, the decoy generation and decoy system may be included in the P2P software being executed by a processing device. The system may be included in the software operating on all processing devices in the P2P network or may be executed by certain processing systems in the network.

The decoy data generation and distribution system operates in the following manner in accordance with this invention. A processing device executing P2P software including the instructions for a system in accordance with this invention connects to a P2P network. The processing device then requests a work from a second processing device or peer on the network. The data for the work is then transferred from the second processing device to the processing device.

When the data is received, the instructions for the system are executed which identify the received work. The system then determines whether the received work is a protected work. This may done by comparing the identity to a list of protected works stored in the processing system or by transmitting the identity to another processing device for determination. If the work is determined to be a protected work, the system alters the data stored in the file of the work to generate decoy data. The file is then stored in memory by the processing system. All identification information for the transferred work is also stored and used for making the file with the decoy data available to other processing devices on the P2P network.

BRIEF DESCRIPTION OF THE DRAWINGS

The above and other advantages and features of this invention are described in the detailed description given below and the following drawings:

Figure 1 illustrating a Peer to Peer network in accordance with this invention;

Figure 2 illustrating a processing system executing instructions for providing a decoy data generation and distribution system in accordance with this invention;

Figure 3 illustrating a flow chart of an exemplary process for establishing a Peer to Peer network connection in accordance with this invention;

Figure 4 illustrating a flow diagram of a process for providing the generation and distribution system in accordance with this invention; and

Figure 5 illustrating a flow diagram of a process for inserting data into an audio file in accordance with an exemplary embodiment of this invention.

DETAILED DESCRIPTION

This invention relates to a system for generating decoy data and distributing the decoy data over a Peer to Peer network. The following is a description of exemplary embodiments in accordance with this invention. Where appropriate, components shown on different figures are given the same reference numeral throughout the description.

Figure 1 illustrates a Peer to Peer connection 150 between a first processing device 105 and a second processing device 110. First processing device 105 is connected to network 100 via path 115. Path 115 may be a telephonic, wireless or other connection to a processing device (Not Shown) in network 100. Second processing device 110 is connected to network 100 via path 120. Path 120 may be a telephonic, wireless or other connection to a processing device (Not Shown) in network 100. Both first processing device 105 and second processing device 110 store Peer to Peer software in a memory. Peer to Peer connection 125 is made over network 100 by both first processing system 105 and second processing 110 executing the peer to peer software store in the respective memories of the processing systems.

Figure 2 illustrates an illustrative exemplary embodiment of a processing system. A processing system is a computer or other processing device that is capable of executing instructions to provide an application. One skilled in the art will recognize that the exact configuration of a processing device executing instructions to provide a system in accordance with this invention may vary depending on the design specification of a

particular processing system.

Processing system 200 has a Central Processing Unit (CPU) 201. CPU 201 is a processor, microprocessor, or any combination of processors and/or microprocessors that execute instructions stored in memory to perform an application. CPU 201 is connected to a memory bus 203 and Input/Output (I/O) bus 204.

A non-volatile memory such as Read Only Memory (ROM) 211 is connected to CPU 201 via memory bus 203. ROM 211 stores instructions for initialization and other systems command of processing system 200. One skilled in the art will recognize that any memory that cannot be written to by CPU 201 may be used for the functions of ROM 211.

A volatile memory such as Random Access Memory (RAM) 212 is also connected to CPU 201 via memory bus 203. RAM 212 stores instructions for all processes being executed and data operated upon by the executed processes. One skilled in the art will recognize that other types of memories such as DRAM and SRAM may also be used as a volatile memory and that memory caches and other memory devices (not shown) may be connected to memory bus 204.

Peripheral devices including, but not limited to, memory 221, display 222, I/O device 223, and network connection device 224 that are connected to CPU 201 via I/O bus 204. I/O bus 204 carries data between the device and CPU 201. Memory 201 is a device for storing data unto a media. Some examples of memory 221 include read/write compact discs (CDs), and magnetic disk drives. Display 222 is a monitor or display and associated drivers that convert data to a display. I/O device 223 is a keyboard, a pointing device or other device that may be used by a user to input data. Network device 224 is a modem or Ethernet "card" that connects processing system 200 to a network..

This invention relates to a system that generates and distributes decoy data for a work over a Peer to Peer network. A Peer to Peer network is a group of loosely coupled processing systems that communicate with one another to exchange files. There are many different architectures or configurations for Peer to Peer networks. The exact architecture of a Peer to Peer network is not important to providing a system in accordance with this invention. Figure 3 illustrates an illustrative process executed by a processing system to connect to a Peer to Peer network and exchange data. One skilled in the art will appreciate that process 300 is an illustrative process and other processes may be used to connect to a Peer to Peer network and exchange data in accordance with this invention.

Process 300 begins in step 305 with the Peer to Peer software being initiated on a processing device. In step 310, the processing unit sends out a request for other processing units connected to the network and executing the Peer to Peer software to respond. In some embodiments, the processing system may transmit non-routable request over the network. In some other embodiments, the processing system may transmit a routable request over the network. In still other embodiments, the executed software includes an address on the network to contact when connecting to the Peer to Peer network. The address may be an IP (Internet Protocol) address, a URL, or other addressing protocol for transmitting data to another processing system connected to the network.

In step 315, a reply is received from a processing system connected to the network and executing the Peer to Peer software. The reply includes address information for other processing systems connected to the network and executing the peer to peer software. This address information may be an IP address, URL, or other resource location information.

In step 320, the processing system reads the list of address information and stores the list in memory. The address information may be used by the processing system to contact other processing systems in the Peer to Peer network. This list may then be sent be the processing system when another processing system in the Peer to Peer network contacts the processing system. This allows the processing system to immediately participate in the massive distribution of address information of active peers to allow all processing systems in the peer to peer network to locate and connect with other processing systems in the network. Furthermore, this allows the processing system to quickly rejoin the network the next time the processing system connects or if the processing system where to be disconnected from the network.

The processing system is now connected to the Peer to Peer network. In step 325, the processing system then compiles a list of content stored by the processing system that is available to other system on the network. In a typical network, only a descriptor of the content of a file available to share is used. However, the list may include content titles, artists, source, encoding method, genre, category, software manufacturer, right holder, and other information. In addition, this information may include a SHA-1 hash, MD5 hash, TorrentID, or other identifier that uniquely describes the content of a file available to share over the network. One skilled in the art will recognize that the exact information is left to a designer of the Peer to Peer network.

In step 330, the list is then transmitted to all processing system identified on the list of connected processing systems stored in memory. One skilled in the art will recognize that list may only be stored locally in some embodiments of peer to peer networks.

In step 335, the processing system receives lists of available contents from other processing systems in the Peer to Peer network. In step 340, the address information for the processing system from which the list was received is stored. After a list is received from a processing system, the list of contents is merged into a list of available content stored by the processing system in step 340. In step 345, the processing system may transmit the list of available content to another processing system in the network. This allows information about available content to be quickly spread to other system connected to the network.

In step 347, the processing system receives a request for content available over the network. The request may be an input from a user. The request may also be a result of an automated process that is searching for a certain content. In step 350, the processing system searches the list of available material to determine a processing system in the network that has the content available. One skilled in the art will recognize that in some systems more than one processing system storing the desired content may be contacted. However, only a second processing system is used in this example to better show the file transfer process.

In step 355, the processing system then establishes a connection with a second processing system providing the content over the network. After the connection is established, the processing system requests the content from the second processing system. One skilled in the art will recognize this request may include some authentication procedure in which the processing system requesting content must transmit authentication data to the second processing system.

In step 360, the processing system receives authorization to receive the content from the second processing system. In some systems, the second processing system may also send a denial of the transfer or an indication that the requested content is no longer available from the second processing system. In step 365, the processing system receives the requested content from the second processing system and process 300 ends.

One skilled in the art will recognize that since the list of systems on the system may change constantly that many methods may be needed to assure that a connection to a processing system providing the requested content and minimize network traffic. However, these are outside the scope of the current discussion and not important to understanding this invention.

This invention relates to the generation and distribution of decoy content for a work over a Peer to Peer network. Instead of introducing decoy content by pretending to be a peer processing system connected to the network, the decoy data for the content of the work is inserted into a file storing the content after the file has been transferred over the network. The insertion is performed by the Peer to Peer application that provides the connection to the network. The instruction for a system in accordance with this invention may be stored as software, firmware, hardware, or any other tangible medium readable by a processing system to execute the instruction to perform processes.

Process 400 illustrated in Figure 4 is an exemplary process for inserting decoy data into a file containing data for a protected work in accordance with this invention. Process 400 begins in step 400 by receiving the data of a work over a Peer to Peer network. The data may be in a data file or in some other form. In step 410, the processing system reads the data. In step 415, the work is identified by the read data. The data maybe identified using any of a variety of techniques known in the art. One method for identifying a work is described in U.S. Patent Number 5,918,223 issued to Blum et al. One skilled in the art will recognize that other process for identifying a work may be used. One example of another method that is use is CDDB.

Once the identity is ascertained, the processing system compares the identity of the work to a list of known protected works in step 420. If the work is found on the list of protected works, the processing system creates decoy content in the data in step 425. An exemplary embodiment of the processing system of creating the decoy content is described below in process 500, shown in Figure 5. After the decoy content is stored in the data or if the work is not on the list of known protected works, the data is stored to memory in step 430.

In step 435, the processing system stores identifiers of the undamaged data. Thus, when the processing system generates a list of available content, the identifiers for the undamaged data of the work is used. This allows the decoy content for the work to be available for transfer with the information of the undamaged copy. This allows the decoy version to defeat some processes for detecting decoy material in Peer to Peer networks.

In step 440, the processing system generates a new set of identifiers for the data including the decoy content and stores the new set of identifiers in step 445. This allows the processing system to make the decoy available under a new identifier to try to prevent detection of the decoy material. The generating of identifiers may include generating a new

title, a new hash code, and/or other identifiers used to identify the content of the data.

In step 450, the processing system then makes the stored decoy data in the received file available to other processing systems connected to the Peer to Peer network and process 400 ends. When another processing system connected to the Peer to Peer network searches for a work, that processing system might locate the damaged copy of the work being shared by the processing system. The searching processing system might transfer this decoy content from the processing system to itself. Upon receiving the decoy content, the searching processing system may note the damage and discard the content in which case the P2P network has become less useful. However, upon receiving the decoy content, the searching processing system may fail to notice the damage and keep the decoy in which case the searching processing system will, in turn, share this decoy content back into the network.

The decoy data inserted into the received data may be created by a variety of techniques as commonly known in the art. In the case of an audio work, the work might have a voiceover inserted imploring the listener to "please purchase a legitimate copy". In the case of an image, the work might have a light overlay of lettering dominating the image. In the case of video, the work might be damaged by having both bold lettering inserted in the center of each frame and an imploring voiceover. In the case of software, the work might have a few key bits flipped to prevent reliable operation or bits inserted that cause the software to display a "please buy me" message when it is executed. In the case of a document, some of the content might be scrambled. The part of the content that is modified is not important. What is important is that the particular copy of the work is rendered less valuable in a way that satisfies the owner of the copyright of the work.

Figure 5 illustrates steps in an exemplary process 500 for inserting decoy data into data of an audio work. One skilled in the art will recognize that there are many other methods for inserting voice over and other types of data into data for an audio work that may be implemented. One skilled in the art will further recognize that other types of processes may be necessary for other types of works including images and audio/visual works.

Process 500 begins in step 505 with the data for the work being read. In step 510, the content of the audio work is converted into a Pulse Code Modulation (PCM) format. In step 515, the data is altered. The re-encoding may include reducing the volume of the content by 70% and/or inserting a voice over. Another method of re-encoding may include replacing audio frames directly. The replacement causes an abrupt transition from the work to the inserted data. After the audio data is altered, the PCM data is re-encoded in a form

suitable for storage in step 520 and process 500 ends.

In this way each peer on the network becomes a source of decoy content. There is no need for a copyright holder to employ massive spoofing technology to flood the network. As peers naturally migrate to different IP addresses through the normal mechanisms well known in the art, it will be difficult for the P2P network to excise the peer or the content from the network. Essentially the network will become unable to trade undamaged content unless the copyright owner allows it.

CLAIMS

I claim:

1. A method for providing decoy data for a work in a peer to peer network:

receiving data for a work over a peer to peer connection;

determining whether said work is a protected work;

inserting decoy data into said data of said work responsive to a determination said work is a protected work;

storing said data including said decoy data of said work in a memory.

2. The method of claim 1 further comprising:

making said data including said decoy data of said work available for transfer over said peer to peer network.

3. The method of claim 1 further comprising:

storing said identifier information of said data of said work received over said peer to peer connection.

4. The method of claim 3 further comprising:

making said data of said work including said decoy data available for transfer over said network using said identifier information of said data of said work received over said peer to peer connection.

5. The method of claim 1 further comprising:

generating identifier information for said data of said work including said decoy data; and

storing said generated identifier information in said memory.

6. The method of claim 5 further comprising:

making said data of said work including said decoy data available for transfer over said peer to peer network using said generated identifier information.

7. The method of claim 1 further comprising:

determining an identity of said work responsive to receiving said data of said work.

8. The method of claim 7 wherein said step of determining whether said work is a protected work comprises:

comparing said identity of said work to a list of protected works responsive to a determination of said identity of said work; and

determining said work is a protected work responsive to said identity being

on said list of protected works.

9. The method of claim 1 wherein said work is an audio work and said step of inserting decoy data comprises:

inserting a voice over into said data.

10. The method of claim 1 wherein said work is an audio work and said step of inserting decoy data comprises:

reducing the volume of said work.

11. The method of claim 1 wherein said work is an image and said step of inserting decoy data comprises:

inserting an overlay that dominates said image.

12. The method of claim 1 wherein said work is a video work and said step of inserting decoy data comprises:

inserting lettering in each frame of said video.

13. The method of claim 1 wherein said work is a video work and said step of inserting decoy data comprises:

inserting audio into an audio portion of said video.

14. An apparatus for providing decoy data for a work in a peer to peer network:

circuitry configured to receive data for a work over a peer to peer connection;

circuitry configured to determine whether said work is a protected work;

circuitry configured to insert decoy data into said data of said work responsive to a determination said work is a protected work;

circuitry configured to store said data including said decoy data of said work in a memory.

15. The apparatus of claim 14 further comprising:

circuitry configured to make said data including said decoy data of said work available for transfer over said peer to peer network.

16. The apparatus of claim 14 further comprising:

circuitry configured to store said identifier information of said data of said work received over said peer to peer connection.

17. The apparatus of claim 16 further comprising:

circuitry configured to make said data of said work including said decoy data available for transfer over said network using said identifier information of said data of said work received over said peer to peer connection.

18. The apparatus of claim 14 further comprising:

circuitry configured to generate identifier information for said data of said work including said decoy data; and

circuitry configured to store said generated identifier information in said memory.

19. The apparatus of claim 18 further comprising:

circuitry configured to make said data of said work including said decoy data available for transfer over said peer to peer network using said generated identifier information.

20. The apparatus of claim 14 further comprising:

circuitry configured to determine an identity of said work responsive to receiving said data of said work.

21. The apparatus of claim 20 wherein said circuitry configured to determine whether said work is a protected work comprises:

circuitry configured to compare said identity of said work to a list of protected works responsive to a determination of said identity of said work; and

circuitry configured to determine said work is a protected work responsive to said identity being on said list of protected works.

22. The apparatus of claim 14 wherein said work is an audio work and said circuitry configured to insert decoy data comprises:

circuitry configured to insert a voice over into said data.

23. The apparatus of claim 14 wherein said work is an audio work and said circuitry configured to insert decoy data comprises:

circuitry configured to reduce the volume of said work.

24. The apparatus of claim 14 wherein said work is an image and said circuitry configured to insert decoy data comprises:

circuitry configured to insert an overlay that dominates said image.

25. The apparatus of claim 14 wherein said work is a video work and said circuitry configured to insert decoy data comprises:

circuitry configured to insert lettering in each frame of said video.

26. The apparatus of claim 14 wherein said work is a video work and said circuitry configured to insert decoy data comprises:

circuitry configured to insert audio into an audio portion of said video.

Figure 1

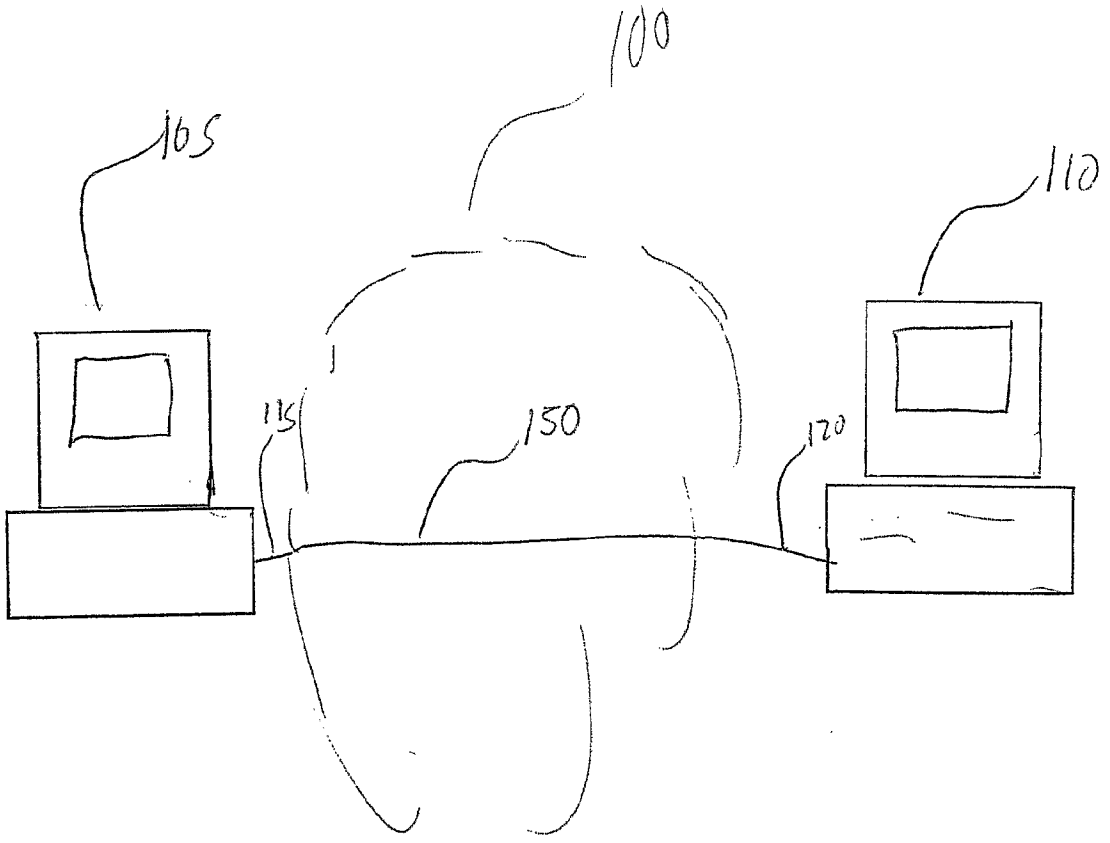


Figure 2

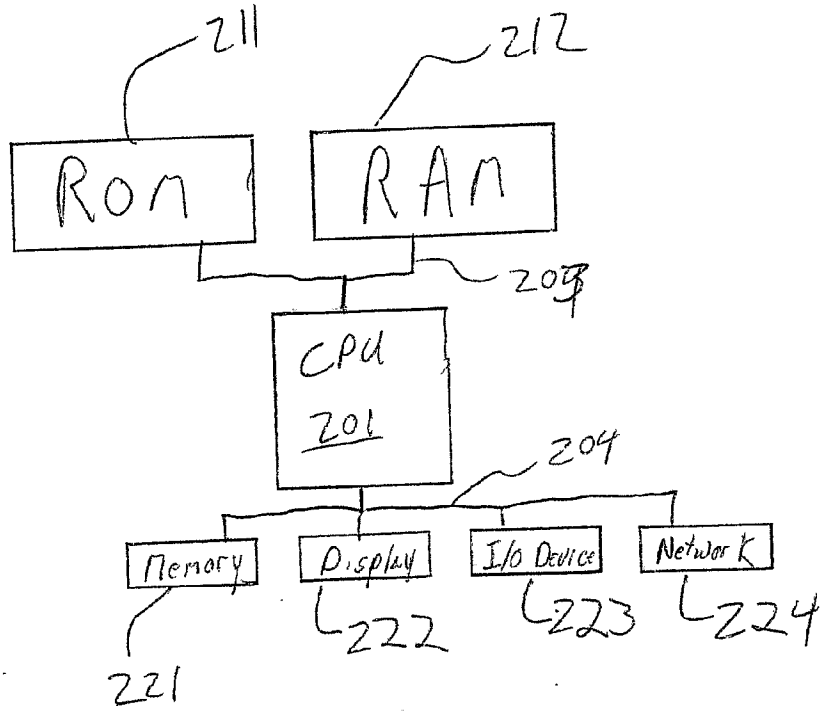

211
212
ROM
RAM
209
CPU
201
204
Memory
Display
I/O Device
Network
221
222
223
224

Figure 3

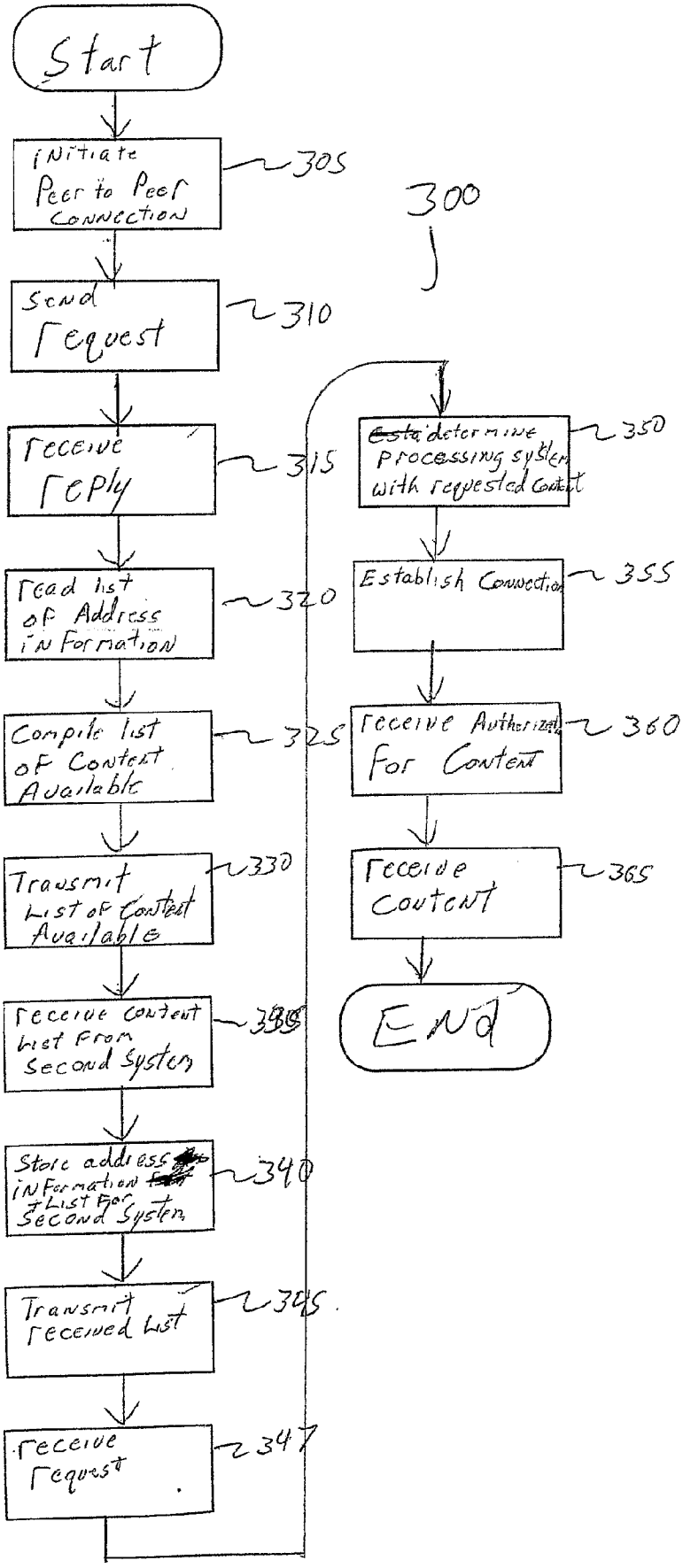

Start
Initiate Peer to Peer Connection
305
300
Send Request
310
Receive Reply
315
Read list of Address information
320
Compile list of Content Available
325
Transmit List of Content Available
330
Receive Content List From Second System
335
Store address information List for Second System
340
Transmit received List
345
Receive Request
347
determine Processing system with requested Content
350
Establish Connection
355
Receive Authorization For Content
360
Receive Content
365
End

Figure 4

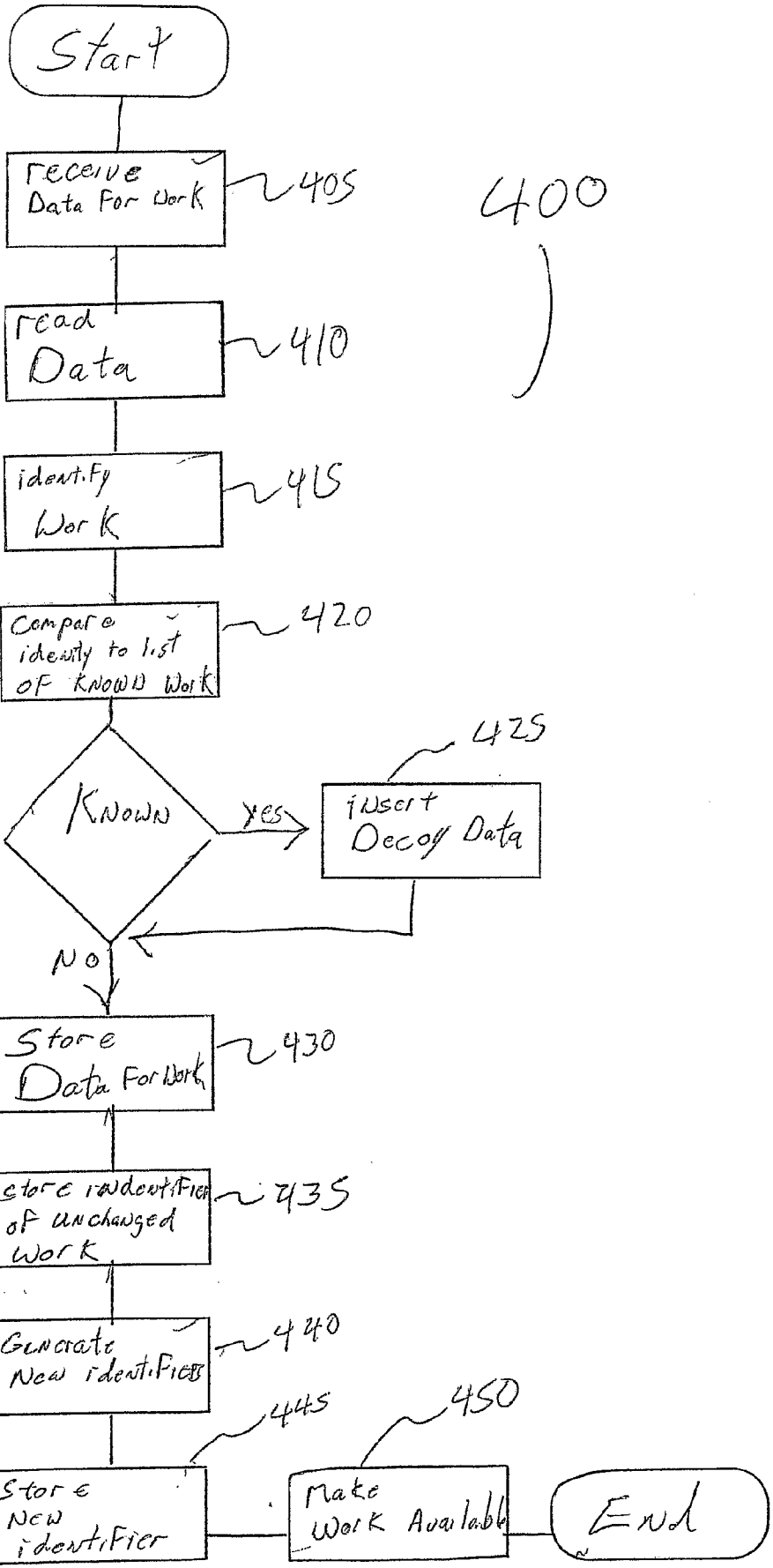
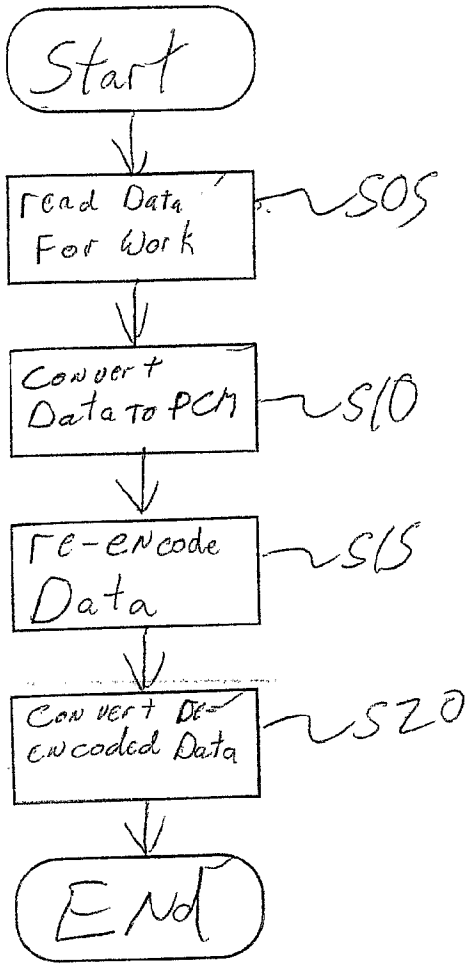

Start
Receive Data For Work
405
400
Read Data
410
Identify Work
415
Compare identity to list of KNOWN Work
420
KNOWN
Yes
425
Insert Decoy Data
No
Store Data For Work
430
Store identifier of unchanged work
435
Generate New identifiers
440
445
Store New identifier
450
Make Work Available
End

Figure 5


Start
Read Data For Work
S05
Convert Data To PCM
S10
Re-encode Data
S15
Convert De-encoded Data
S20
End
500